Server Message Block in the Age of Microsoft Glasnost

CHRISTOPHER R. HERTEL

Christopher R. Hertel is a long-haul member of the Samba Team and co-founder of the jCIFS project. He is also the author of *Implementing CIFS—The Common Internet File System,* the only developer's guide to the SMB/CIFS protocol suite. Not too long ago, he had the opportunity to work directly with Microsoft's File Server team when the company he founded, ubiqx Consulting, Inc., was tapped to write Microsoft's official SMB/CIFS specifications. Chris has also been adjunct faculty at the University of Minnesota College of Continuing Education (CCE) and is currently a member of the CCE IT Infrastructure Advisory Board.

crh@ubiqx.com

The EU anti-trust case against Microsoft concluded in late 2007. Related or not, that's when things started to change. One pleasant surprise for third-party developers was the release of hundreds of specifications covering Windows file formats, system internals, and protocols. Microsoft was opening up. Four years later, about 400 specifications have been published. It took a while for some of those documents to appear, mostly because Microsoft didn't actually have them all written yet. And now they have surprised us again. Well before the beta release of Windows 8, they have provided preview documentation for an overhauled and compelling new version of the venerable Server Message Block Protocol: SMB2.2. This is going to be epic.

A Gathering of Storage Geeks

Once a year, typically in September and typically somewhere near San Jose, California, the Storage Networking Industry Association (SNIA) hosts the Storage Developer Conference (SDC). The season and location don't really matter all that much, since many of those who attend never leave the confines of the hotel until the conference is over. As its name implies, the SDC is aimed at data storage engineers.

As a sideshow to the SDC, the SNIA hosts the annual SMB/CIFS/SMB2 Plugfest, an opportunity to bring together different products from different vendors and open source groups and do some heavy-duty interoperability testing.

The Plugfest is housed in its own conference room. The room is filled with network cables and switches, computers, tables with black tablecloths, a whiteboard, racks of storage equipment (with fans that sound like jet engines), and engineers. A badge is required to get in, but for those who participate there are only a few simple rules: Don't take stuff that's not yours, don't talk about your competitor's test results outside the Plugfest room, but do take a shower at least once every two days. They're serious about that last rule. They need to be. It gets intense in there.

The Ice Age

The Plugfest has been an annual event for more than a dozen years—longer, in fact, than the SDC—but for most of those years Microsoft, the "owner" of the SMB protocol suite, was conspicuous by their absence. While they were in the throes of legal battles (with not one but two major world governments) they kept their SMB code monkeys locked in their cages in Redmond.

During that time, SMB also suffered a dearth of documentation. There were some vintage OS/2 and DOS docs, and an unpolished, incomplete draft specification that Microsoft had presented to the IETF in 1996/97. The SNIA made a valiant attempt to create an updated version of the IETF draft, but without a solid commitment from Microsoft it too suffered from inaccuracies and omissions. There was also an implementer's guide (written by yours truly) that leveraged the collected knowledge of the third-party SMB development community. Since Microsoft's engineers weren't allowed out to play with the rest of the development community, and since there were no official specifications, SMB remained a protocol shrouded in myth and obscurity.

To make matters worse, when Microsoft lost in the US courts they furthered the divide by creating a licensing program that provided some of their competitors with access to Microsoft-internal information. That licensing program originally required signing an NDA (and possibly paying a fee, but we don't know for sure, because that part fell under the NDA). A couple of companies opted in, several others opted out, and open source groups such as the Samba Team were left without any options. The licensing program effectively split the SMB development community.

Those were the dark times—ash fell from the sky, a biting frost covered the land, the sun was only seen dimly behind the clouds, and SMB developers became melodramatic and prone to hyperbole. The Plugfest almost came to an end.

After Microsoft conceded in the EU in late 2007, things changed again. This time, however, Microsoft surprised just about everyone (who was paying attention) by going in completely the opposite direction. One of the first moves they made was to publicly release hundreds of documents covering Windows file formats, protocols, and software internals. They simply started posting them on the Web. Then they unlocked the cages and let their code monkeys out. Not only did they send engineers to the Plugfest, they went so far as to sponsor it so that others—notably smaller companies and open source groups—could afford to attend as well.

It was like springtime. The ice melted, the sun shone, the divide was healed, and the community started to come back together. In 2008, when Microsoft first sent a contingent to the Plugfest, a total of 14 organizations participated. By 2011 there were 27, almost twice as many.

Quick Sync: SMB and CIFS

Just to make sure we're all on the same page here, the SMB protocol is what Windows systems use to connect the Q: or Z: "drive" to a shared directory on a server—possibly a Windows server or, thanks to those storage engineers at the Plugfest, it could be any of several third-party implementations.

The naming could use a little clarification as well. Just before the start of the SMB Ice Age, when Microsoft submitted the draft specification to the IETF, they gave the protocol a marketing upgrade. That is, they renamed it. They decided that they needed to have the word "Internet" in the acronym, so they re-dubbed it Common Internet File System: CIFS. These days, the CIFS name is used interchangeably with SMB. They essentially mean the same thing.

SMB Evolution

In the computing industry, we measure time in nanoseconds. By that standard SMB is a fossil. It was originally created by IBM (not Microsoft!) in the 1980s as a network file protocol for PC-DOS, yet somehow it survives today as a key compo-

nent of Microsoft's Windows products. Not only is it still around, it is probably the most widely used and most successful network file system there is. Apple AFP? Novell Netware? Gone by the wayside. The only real competition still out there is NFS, which is as old and wizened as SMB.

All Windows systems, of course, support SMB. In addition, every major NAS platform and every major server OS (as well as many of the minor ones) supports SMB. Mainframes and desktops and tablets and cell phones support SMB. It's an interoperability requirement these days. If you find a NAS appliance for the home or enterprise or anything in between that doesn't support SMB, you should buy a lottery ticket—it's your lucky day.

Despite its importance, development of the SMB protocol remained relatively stagnant for years. In fact, SMB hasn't changed much at all since it was ported from OS/2 to Windows NT back in the early 1990s. When Windows 2000 was being developed, there were rumors that it would have a completely overhauled version of SMB to go along with the new CIFS name, but the overhaul didn't actually happen then. Windows 2000 introduced a few enhancements to access additional Windows OS features, improvements to the authentication subsystem, and changes to the entourage of protocols that support SMB. At the core, however, the SMB protocol remained the same.

Somewhere in the deep of the dark times, Microsoft finally decided that they needed something better. So, as part of Windows Vista, they introduced SMB version 2 (SMB2). SMB2 is a complete rewrite. The number of commands supported has been cut to 25% of its predecessor's, a reduction in bulk made possible by consolidating redundant commands and (hurray!) jettisoning support for DOS and OS/2 semantics. SMB2 also offers some useful new features, though none of those features are immediately visible to the user; SMB2 is a transparent upgrade. It is so transparent, in fact, that users and system administrators often don't even know it's there. It runs automatically instead of SMB if both the client and server agree to use it.

So SMB2 was a major cleanup but not a paradigm shift. A few additional features were added in SMB version 2.1, which appeared with Windows 7, but those were seen as incremental improvements. One major step forward, however, was documentation. Windows 7 came out after Microsoft had started to open up. That, and the realization that Windows XP was the only supported Windows OS that did not include SMB2 encouraged third-party developers to start thinking of SMB version 1 as the old reliable family minivan—it still runs, but it's too big and it's getting a little rusty around the edges. Time to move up to something sportier and more efficient, but we'll keep the old one parked in the back until it stops running entirely.

Call Me Triceratops

There are several presentation tracks at the SDC, covering all manner of storage technologies, from the mechanics of disk drives to the problems of defining standards for content-addressable archive metadata. There are also, of course, keynote speeches. At the 2010 SDC, a keynote slot opened up at the last minute and the organizers asked Samba Team luminary Jeremy Allison to fill the gap. Never fear, Jeremy's here. Somehow, he quickly threw together a remarkably memorable talk in which he asserted that the NFS and SMB developers in the room, himself included, were a herd of dinosaurs all trying to eat one another before the asteroid collision and the rise of the rodents.

Jeremy pointed out that most of the new and interesting work in network file systems was being done to create targeted solutions like Gluster, Hadoop HDFS, Ceph, and any number of other efforts aimed at solving specific sets of problems rather than trying to do a good enough job at everything to handle every use-case. General-purpose network file systems, he asserted, were a dying breed. A handful of them are still out there, but (as noted above) the mainstream market has narrowed its interest down to just two: NFS and SMB.

The Age of the Dinosaurs

During the dark times, while Microsoft was hibernating, the Internet Engineering Task Force (IETF) was busy overhauling NFS and doing so out in the open as part of a standards process. NFSv4 and 4.1 were getting snazzy new features such as parallel data access and transport over RDMA. SMB was getting left behind.

That was something else that changed when Microsoft let their engineers out to play with the other engineers. They saw the bright shiny things going into NFS, and they saw that the Samba Team had somehow managed to implement SMB clustering. Perhaps Microsoft had been planning an advanced feature set for SMB2 from the very start, but it wasn't until they announced version 2.2 at the 2011 SDC that it became clear that they, too, had been hard at work re-evaluating the market for general-purpose network file systems.

One year after Jeremy's keynote (which they made a point of acknowledging), Microsoft gave their own SDC keynote titled "The Future of File Protocols: SMB 2.2 in the Data Center." The presentation was full of surprises. In addition to providing an overview of new SMB2.2 features in Windows 8 and giving a live demo that actually worked, they also announced the availability of preview documentation so that others could start building implementations right away.

Here's some of what will be in SMB2.2:

◆ SMB2.2 supports clustering: Multiple SMB servers can be connected to multiple application servers, providing scalability as well as failover.
◆ SMB2.2 supports multi-path: There can be multiple connections between the client and the server.
◆ SMB over RDMA: SMB2.2 I/O commands can use machine-to-machine remote direct memory access (RDMA) to perform read/write operations over iWarp, InfiniBand, or plain old Ethernet.
◆ SMB2.2 provides lightweight WAN acceleration: This feature, called Branch-Cache, was introduced in SMB2.1, but needs to be enabled in order to be used. SMB2.2 supports an updated version of BranchCache that allows for improved content caching algorithms.

The most surprising thing about these features is that they were demonstrated live using a pre-alpha version of Windows 8 that had been compiled the night before by a guy named Dave, and the system never crashed. (It's also surprising that we know Microsoft SMB developers by name, even if only first name. That was never the case back in the dark times.) It's hard enough to get a live demo working with production code, so using pre-alpha code is downright scary. Just to make things more fun, they were connecting and disconnecting cables during the keynote to show how well the system failed over and scaled.

It was a very impressive demonstration, and fun to watch in the way that a daredevil act is fun to watch. For Windows users, SMB2.2 will be a major—and

visible—step forward. However, a quick look through the list of key features shows that there is nothing actually new here, technology-wise. SMB2.2 is running fast to catch up.

The Battle for the Niche

So if Jeremy is right, what we are about to witness is a battle for dominance between two large tyrannosaurs in what remains of the general-purpose file protocol ecosystem. NFS is ahead on features, but NFS implementers have been slow to create products, and users have quite reasonably been slower to adopt them. SMB has the advantage of Windows 8—when that comes out, both clients and servers will have SMB2.2 already installed and ready to deploy. It will simply "be there."

What this means for developers is that there is a lot of work to be done. On the NFS side, more v4 implementations need to become available on more platforms. Soon. On the SMB side, there are not enough third-party SMB2 implementations, and those that do exist will need to be updated to handle the fancy SMB2.2 feature set.

And there is one more logistical difference to consider. The rumor among Silicon Valley recruiters is that you can put up a sign saying "NFS Coders Needed" and a line will form, but that there simply aren't enough SMB developers to be found. Universities often teach basic NFS internals, and the specifications have long been available, so NFS know-how is a relatively common skill.

Not so with SMB. Most SMB developers are either senior engineers who have spent years working with the protocol and have learned it from the ground up, or senior engineers who became senior working on some other protocol and were then tapped by their employer to learn SMB. There are a few, but very few, newcomers in the SMB world.

Place your bets, the stakes are high. Data storage isn't the most glamorous field in modern computing, but it is fundamental. What happens in this arena in the next few years will have implications. Rodents take note: the dinosaurs still rule.

Resources

Christopher R. Hertel, *Implementing CIFS—The Common Internet File System* (online book), 2003: http://www.ubiqx.org/cifs/#Contents.

Christopher R. Hertel, 2011 SNIA SDC Report: Summary (blog post), Samba Team: http://www.samba.org/samba/news/developers/2011-snia-sdc-report.html.

Microsoft Corporation, [MS-CIFS]: Common Internet File System (CIFS) Protocol Specification, 2009: http://msdn.microsoft.com/en-us/library/ee442092.aspx.

Microsoft Corporation, [MS-SMB]: Server Message Block (SMB) Protocol Specification, 2007: http://msdn.microsoft.com/en-us/library/cc246231.aspx.

Microsoft Corporation, [MS-SMB2]: Server Message Block (SMB) Version 2 Protocol Specification, 2007: http://msdn.microsoft.com/en-us/library/cc246482.aspx.

Microsoft Preview Specifications (includes SMB2 updated to include SMB2.2 specifications; SMBD, which specifies SMB2 over RDMA Transport Protocol; MS-SWN: SMB2 Witness Protocol Specification; MS-SWN cluster failure/recovery notification): http://msdn.microsoft.com/en-us/library/ee941641.aspx.